

Le dernier travail pratique se fera dans le cadre du concours du meilleur algorithme. L’objectif est de concevoir un algorithme de votre cru pour résoudre un problème combinatoire. Le classement des équipes déterminera le nombre de points accordés pour la qualité de l’algorithme.

## 1 Problématique

Un nouveau parc va ouvrir dans les Laurentides et vous êtes mandaté pour décider du tracé des chemins de randonnées. Le parc a déjà décidé des points d’intérêt qui vont constituer le parc et leurs positions sont connues. Il y a trois types de points d’intérêt sur leur territoire :

- Type 1 : Les points de vue époustouffants
- Type 2 : Les points d’entrée du parc
- Type 3 : Les points étape

Le parc vous demande de tracer les sentiers pour accéder aux points d’intérêt tout en respectant certaines contraintes :

1. Pour chaque point d’intérêt, il doit exister un chemin qui le relie à une entrée du parc.
2. Chaque entrée de parc doit être le départ d’au moins un sentier.
3. Les points étape doivent avoir au moins 2 sentiers incidents.
4. Les points de vue ne sont accessibles que par un seul sentier.
5. Chaque point d’intérêt possède un nombre limité de sentiers incidents.

À chaque sentier possible reliant deux points d’intérêt est associé un coût qui correspond aux coûts des travaux d’aménagement du chemin.

Votre rôle est de décider d’un tracé des chemins de randonnée qui répond aux contraintes et qui a un coût total minimal pour le parc.

## 2 Jeu de données

Des exemplaires sont donnés sur Moodle. Différentes variantes de parc sont proposées. Les fichiers sont nommés ainsi :

Parc[*no de la variante*]-[*nb de points d’intérêts*]Zones.txt

Le format d’un exemplaire est le suivant :

```
<nb de points d’intérêt (n)>
<liste des types des points d’intérêt>
<liste du nombre maximum de sentiers associés à chaque point d’intérêt>
<matrice n x n indiquant les coûts entre chaque paire de sommets>
```

Le code pour les types de points d'intérêt est le suivant :

- 1 : Point de vue époustouflant
- 2 : Entrée du parc
- 3 : Point étape

Les points d'intérêt sont implicitement numérotés de 0 à n-1.

### 3 Résultats

Il est d'une grande importance de respecter le format de sortie, puisque l'évaluation de la qualité de votre algorithme passera d'abord par le test de l'intégrité de la solution. Ce dernier échouera si votre format de sortie n'est pas respecté et votre programme serait considéré non fiable ce qui entraîne une note de 0 sur 4 pour la qualité de l'algorithme. À cet effet, on vous suggère fortement de commencer tout d'abord par implanter un petit programme qui vérifie l'intégrité de vos solutions, ainsi au fur et à mesure que vous apporterez des modifications à votre algorithme, votre programme fera le test pour s'assurer que les solutions sont valides.

Lors de l'évaluation, votre programme sera exécuté sur plusieurs exemplaires de notre choix. Nous utiliserons alors les ordinateurs du laboratoire L-4714 pour tester vos algorithmes. Nous considérerons la dernière solution affichée après 3 minutes de calcul. Dès qu'une solution (sous-optimale) a été trouvée, affichez-la à la sortie standard dans le format décrit plus bas ; votre programme sera interrompu une fois la limite de temps atteinte s'il n'a pas encore terminé.

### 4 Remise

Vous devez remettre votre travail sur Moodle au plus tard le mercredi 19 avril à 23h59. Votre rapport doit comprendre :

1. une brève description du sujet et des objectifs de ce travail ;
2. une présentation de votre algorithme sous forme de pseudo-code et incluant une analyse de complexité théorique des principales fonctions ;
3. une justification de l'originalité de votre algorithme ;

Le rapport pour ce dernier travail pratique est assez succinct. Nous vous encourageons à terminer ce travail assez tôt afin de ne pas compromettre votre préparation à vos examens finaux.

Votre remise doit être faite dans un répertoire qui porte le nom suivant :

INF4705\_TP3\_matricule1\_matricule2

Le rapport doit être sous format PDF.

L'exécutable de votre programme doit être présent à la racine du projet et testé fonctionnel sur les ordinateurs du laboratoire L-4714.

## 5 tp.sh

Utilisation

```
tp.sh -e [path_vers_exemplaire]
```

Arguments optionnels :

- p Affiche la solution dans le format spécifié à la section 6
- t Affiche le temps d'exécution

Des exemples de scripts sont disponibles sur moodle.

## 6 Format de sortie

```
<id d'un point d'intérêt (sentier 0)> <id d'un point d'intérêt (sentier 0)>
.
.
.
<id d'un point d'intérêt (sentier k)> <id d'un point d'intérêt (sentier k)>
fin
```

Chaque ligne représente un sentier, les identifiants des points d'intérêt commencent à 0, l'ordre dans lequel sont indiqués les sentiers n'importe pas.

Exemple de sortie avec 7 points d'intérêt : 0-1(entrée), 2(étape), 3-6(point de vue) :

```
0 2
2 3
1 6
0 4
2 5
fin
```

## 7 Barème de correction

- 1 pt** : exposé du travail pratique
- 2 pts** : description de votre algorithme
- 3 pts** : conception de l'algorithme (lien avec le contenu du cours, originalité, initiatives...)
- 4 pts** : qualité de l'algorithme (classement)
- 2 pts** : les programmes (corrects, structurés, commentés,...)
- 2 pts** : présentation générale et qualité du français